

CLAIMS

What is claimed is:

- 1 1. A computer-implemented method comprising:
2 starting a debug node from a remote node, wherein the debug node is one of a
3 plurality of nodes within an instance of application servers;
4 isolating the debug node from a messaging service of the instance of application
5 servers;
6 debugging an application on the debug node from the remote node; and
7 stopping the debug node from the remote node.
- 1 2. The method of claim 1, wherein starting the debug node further comprises:
2 isolating the debug node from a load-balancing mechanism of the instance of
3 application servers.
- 1 3. The method of claim 2, wherein isolating the debug node from the messaging service
2 of the instance of application servers further comprises:
3 removing the debug node from a broadcast destination list of the instance;
4 instructing the debug node to transition to a debugging state; and
5 accessing a debug port of the debug node to start a debugging session.
- 1 4. The method of claim 3, wherein the accessed debug port is a Transmission Control
2 Protocol (TCP) port of the debug node.
- 1 5. The method of claim 3, wherein debugging the application on the debug node from
2 the remote node comprises debugging a Web application.

1 6. The method of claim 5, wherein debugging the Web application comprises:
2 starting a Web browser on the remote node to interact with the Web application; and
3 receiving a debug notification from the debug port of the debug node.

1 7. The method of claim 6, wherein stopping the debug node from the remote node
2 comprises:
3 ending the debugging session; and
4 disabling the debug node from the remote node.

1 8. The method of claim 1, further comprising,
2 restarting the debug node.

1 9. The method of claim 1, wherein the remote node is a development client.

1 10. The method of claim 1, wherein the debug node is a Java 2 Enterprise Edition server
2 node.

1 11. An apparatus comprising:
2 a development environment to debug a remote application; and
3 a processor and logic executable thereon to
4 start a debug node, wherein the debug node is one of a plurality of nodes
5 within a remote instance of application servers;
6 isolate the debug node from a messaging service of the instance of application
7 servers;
8 debug an application on the debug node; and
9 stop the debug node.

1 12. The apparatus of claim 11, wherein the processor and logic executable thereon to start
2 the debug node further comprises a processor and logic executable thereon to
3 isolate the debug node from a load-balancing mechanism of the instance of
4 application servers.

1 13. The apparatus of claim 12, wherein the processor and logic executable thereon to
2 isolate the debug node from the messaging service of the instance of application servers
3 further comprises a processor and logic executable thereon to
4 remove the debug node from a broadcast destination list of the instance;
5 instruct the debug node to transition to a debugging state; and
6 access a debug port of the debug node to start a debugging session.

1 14. The apparatus of claim 13, wherein the processor and logic executable thereon to
2 debug the application on the debug node comprises a processor and logic executable thereon
3 to
4 debug a Java-based application executing on a Java virtual machine of the debug
5 node.

1 15. The apparatus of claim 12, wherein the processor and logic executable thereon to stop
2 the debug node comprises a processor and logic executable thereon to
3 end a debugging session between the apparatus and the debug node; and
4 disable the debug node.

1 16. A system comprising:
2 a means for starting a debug node from a remote node, wherein the debug node is one
3 of a plurality of nodes within an instance of application servers;

4 a means for isolating the debug node from a messaging service of the instance of
5 application servers;
6 a means for debugging an application on the debug node from the remote node; and
7 a means for stopping the debug node from the remote node.

1 17. The system of claim 16 wherein the means for starting the debug node further
2 comprises:

3 a means for isolating the debug node from a load-balancing mechanism of the
4 instance of application servers.

1 18. The system of claim 17, wherein the means for isolating the debug node from the
2 messaging service of the instance of application servers further comprises:

3 a means for removing the debug node from a broadcast destination list of the
4 instance;
5 a means for instructing the debug node to transition to a debugging state; and
6 a means for accessing a debug port of the debug node to start a debugging session.

1 19. The system of claim 17, wherein the means for debugging the application on the
2 debug node from the remote node comprises:

3 a means for debugging a Web application on the debug node.

1 20. The system of claim 19, wherein the means for debugging the Web application
2 comprises:

3 a means for starting a Web browser on the remote node to interact with the Web
4 application; and
5 a means for receiving a debug notification from the debug port of the debug node.

1 21. An article of manufacture comprising:
2 an electronically accessible medium providing instructions that, when executed by an
3 apparatus, cause the apparatus to
4 start a debug node from a remote node, wherein the debug node is one of a plurality
5 of nodes within an instance of application servers;
6 isolate the debug node from a messaging service of the instance of application
7 servers;
8 debug an application on the debug node from the remote node; and
9 stop the debug node from the remote node.

1 22. The article of manufacture of claim 21, wherein the instructions that, when executed
2 by the apparatus, cause the apparatus to start a debug node from the remote node further
3 cause the apparatus to
4 isolate the debug node from a load-balancing mechanism of the instance of
5 application servers.

1 23. The article of manufacture of claim 21, wherein the instructions that, when executed
2 by the apparatus, cause the apparatus to isolate the debug node from the messaging service of
3 the instance of application servers cause the apparatus to
4 remove the debug node from a broadcast destination list of the instance;
5 instruct the debug node to transition to a debugging state; and
6 access a debug port of the debug node to start a debugging session.

1 24. The article of manufacture of claim 21, wherein the instructions that, when executed
2 by the apparatus, cause the apparatus to debug the application on the debug node from the
3 remote node cause the apparatus to
4 debug a Web application on the debug node.

1 25. The article of manufacture of claim 24, wherein the instructions that, when executed
2 by the apparatus, cause the apparatus to debug the Web application on the debug node cause
3 the apparatus to
4 start a Web browser on the remote node to interact with the Web application; and
5 receive a debug notification from the debug port of the debug node.

1 26. The article of manufacture of claim 21, wherein the instructions that, when executed
2 by the apparatus, cause the apparatus to stop the debug node from the remote node cause the
3 apparatus to
4 end the debugging session; and
5 disable the debug node from the remote node.